PROGRAM PROTECTION NEWSLETTER
APRIL, 1985

WHAT IS A PROGRAM PROTECTION NEWSLETTER?

It is a monthly newsletter that will deal with program protection schemes of all types. Each month will cover 3 to 5 specific examples of different program protection schemes. Some will be very easy schemes, others may be very difficult. Some issues will cover word processors, data bases, spread sheets and games. Many different programs will be selected and each will be fully explained as to their type of protection.

Each program example used will be for illustrative purposes only. It is not the intention of this newsletter to promote or encourage illegal or unauthorized duplication of any copyrighted program. UNAUTHORIZED DUPLICATION OF A COPYRIGHTED PROGRAM IS ILLEGAL!!

WHY WOULD ANYBODY NEED A PROGRAM PROTECTION NEWSLETTER?

Many programs have a very sophisticated protection scheme and can not be used on some disk drives (MSD, 4040, 8050 etc.). Other programs use errors on the disk that can be very harmful to the disk drive. Many other people just prefer to use one disk for all their programs of one specific type (i.e. utilities).

According to federal copyright law you are allowed to make an ARCHIVAL copy of any program that you have purchased. You are also allowed to modify your program as long as the modifications are essential to the programs operation. Some protection schemes do not allow the program to be directly backed up. They may require the use of sophisticated copy programs or extensive knowledge of MACHINE LANGUAGE to understand. In either case the average user may not be able to obtain an archival copy of his valued programs.

Another group of people just want to know more about their computer and its associated components. They want to know how programs are protected and how their machine operates. They want to know how to modify their own disks and how to protect their own programs.

What ever reason you have for wanting to learn about program protection this newsletter is for you!

PASS THIS COUPON ON TO YOUR FRIENDS WHO WISH TO LEARN MORE ABOUT PROGRAM PROTECTION SCHEMES.

. . . .

SUBSCRIPTION PRICE $35.00 POST PAID IN U.S. ($40.00 FOREIGN)

NAME. . . . .

ADDRESS . . . .

CITY,STATE,ZIP. . . . .

CSM SOFTWARE INC.
P. O. BOX 563
CROWN POINT, IN. 46307

Each month ~~we get a terrific~~ number of requests for various programs. While we cannot honor each request for individual programs, we do try to find similiar programs that may apply to many different requests. This month we will 'drop back' in time and review a number of the more 'basic' types of program protection schemes. These are the mostly the programs written in BASIC that contain bad blocks. Evidently there is still a large group of readers that are still having trouble with this type of program. Those of you that are more advanced may still want to refresh your memory with these types of schemes. If you are still having trouble with programs of this type refer to the PROGRAM PROTECTION MANUAL FOR THE C-64 (VOLUME I).

The program is KIDS ON KEYS, Copyright 1983, Spinnaker Corp., 1983.

TYPE OF PROTECTION: This program will check TRACK 2 SECTOR 15 for an error 23 and TRACK 3 SECTOR 16 for an error 27. Again we see your R/W head getting 'beat-to-death' more than once just to load the program.

HOW TO COPY: Copy the disk and place an error 23 on 2/15 and an error 27 on 3/16, or use a copy program that will place these errors for you.

HOW TO MAKE A WORKING COPY WITHOUT ERRORS ON THE DISK:

This program utilizes some of the early program protection techniques. The error checking is done through the boot program called "SPIN". "SPIN" is a basic program, but when we attempt to examine it with a normal list, we find one line (5) followed by a remark (REM). Cursor up to this line, press RETURN, and list the program. Everything appears to be fine until we reach line 110. From this point on we find that a portion of each line will disappear. This is a result of the programmer's use of multiple delete characters. This technique is covered in the Program Protection Manual Volume I. We can get a look at the program through LOMON. Using the I command at 0801 and following, will reveal the intent of the program. Notice the B-R (Block-Read) instructions that point to tracks 2 and 3. Another way to get a look at this program is to list it to your printer.

The programmer has gone to great lengths to protect his boot program, but left the main program virtually unprotected. Let's get to it. All we have to do is to lift a working version of the program from the computer's memory.

1. Make a copy of the program disk without errors. NEVER, NEVER ATTEMPT TO MODIFY AN ORIGINAL DISK!!!
2. Load and run the program in the normal fashion.
3. Once you reach the main screen, press the RUN/STOP and RESTORE keys.
4. List the program. Cursor up to LINE 1 and press the RETURN key. The entire program is now revealed.
5. You may now save the program to your copy disk with SAVE "KIDS ON KEYS",8.
6. To utilize the program, LOAD "KIDS ON KEYS",8 and RUN.
7. YOU'RE DONE!

The program is PIECE OF CAKE, Copyright 1984, COUNTERPOINT SOFTWARE.

TYPE OF PROTECTION: This program will check TRACK 2 SECTOR 2 for an error 23. If present, the program will run properly.

HOW TO COPY: Use any good copy program capable of placing errors on the destination disk.

HOW TO MAKE A WORKING COPY WITHOUT ERRORS ON THE DISK:

The protection scheme used for this program is similar to the title. Disappearing line numbers is the only form of protection used. When you LOAD the boot program called "PIECE OF CAKE" and attempt to LIST, you will notice that the lines begin to disappear at 40 and continue through 52. The best way to examine this program is to list it to the printer, but it may also be examined through LOMON.

1. Make a copy of the program without errors on the disk, or file copy the programs to another disk.

2. LOAD "PIECE OF CAKE",8. List the program.

3. Line 40 will require a bit of repair, because part of the line is erased when it is listed.

4. Line 40 should read as follows: ONAGOTO73,75,77,99,100,200,202 Don't forget to type RETURN.

5. We will now delete the check for the bad block. Move to the bottom of the screen and type the following:
   ```
   47     (RETURN)
   48     (RETURN)
   49     (RETURN)
   50     (RETURN)
   51     (RETURN)
   52     (RETURN)
   ```

6. You may now save the program to your copy disk with SAVE "@0:PIECE OF CAKE",8.

7. YOU'RE DONE!

The program is ECONOMICS, Copyright 1983, Micro-Ed, Inc.

TYPE OF PROTECTION: The program disk is littered with errors. The program will check for an error and run if present.

HOW TO COPY: Use a copy program that will place the errors for you.

HOW TO MAKE A WORKING COPY WITHOUT ERRORS ON THE DISK.

As with the previous two programs, we are faced with hidden lines. The unprotection process for this program is an easy one, so let's get to it.

1. LOAD and RUN the program in the usual manner. When the MENU appears, choose the first program and allow it to run normally. Once in memory, press the RUN/STOP and RESTORE keys.

2. You may now save the unprotected program to a formatted disk with SAVE "EC1 DOLLAR & CHA",8

3. Repeat STEPS 1 and 2 for the other five programs on the disk.

4. With the six programs saved to disk, we will now alter the menu program so that it may be used without the protection scheme.

5. LOAD "MICRO-READY",8.

6. The lines from 10 through 59090 contain the error protection. Move to the bottom of the screen so that we can delete these lines. Type the following:

```
10          (RETURN)
80          (RETURN)
90          (RETURN)
9999        (RETURN)
59002       (RETURN)
59010       (RETURN)
59020       (RETURN)
59030       (RETURN)
59050       (RETURN)
59060       (RETURN)
59090       (RETURN)
```

7. The menu is now unprotected and may be saved to your formatted disk under any title you wish.

8. YOU'RE DONE!

The program is SPELLICOPTER, Copyright 1983, DesignWare, Inc.

TYPE OF PROTECTION: The program will check TRACK 1 SECTOR 1 for an error 23.

HOW TO COPY: Copy the program with any copy program that can place an error 23.

HOW TO MAKE A WORKING COPY WITHOUT ERRORS ON THE DISK:

Since this program is stored in user files, we will use the back door to unprotect it.

1. Make a copy of the original disk without errors. Format another disk to store the program files we will create.

2. LOAD and RUN the program from the original disk.

3. When the menu screen appears, RESET your computer.

4. LOAD and execute HIMON with SYS 49152.

5. We will not save the new program file to the original disk, because we will run the risk of writing over data that is needed for the other functions of the program (REMEMBER NEVER MODIFY YOUR ORIGINAL DISK). We will save the first section of code to the formatted disk through HIMON.

   Flip out BASIC with your M Command at 0001. Change the 37 to a 36.

   Save the first section of code with S "COPTER1",08,0800,BFFF.

6. Since HIMON wrote over our program code at C000-CFFF, we will have to LOAD the original disk again. Power down and LOAD and execute the original program.

7. When the menu screen appears, RESET your computer. LOAD and execute LOMON with SYS32768.

8. Save the code at C PAGE with S"COPTER2",08,C000,CFFF. Remember, save the code to your formatted disk not the clone disk.

9. Now for the entry point. Using the D command at $0800, we find a JMP to $08DB. This section of code loads in the main menu screen.

10. Power down and load in the two sections of code.

```
LOAD"COPTER1",8,1      (RETURN)
NEW                    (RETURN)
LOAD"COPTER2",8,1      (RETURN)
```

11. With the two sections of code in memory, insert your copy disk and type SYS 2267 RETURN. This is the decimal equivalent for $08DB.

12. YOU'RE DONE!

The only real trick to this program was finding the entry point. A JMP instruction is always a good place to try when looking for an entry point. If a JMP does not work there are other things to look for.

You will find a more complete explanation of entry points in the PROGRAM PROTECTION MANUAL VOLUME II.

When looking for an entry point, be adventurous. You can't harm your computer. You may have to power down and begin again, but all you lose is time. Keep notes on what you have tried. Good notes can save you a lot of time.

You may find that an entry point appears to work, but you get garbage on the screen. This may be the result of your monitor overwriting a section of code needed for the proper execution of the program. Before you begin your hunt for a new entry point, try loading the program you are working on in the normal manner. Stop the program and try your entry point without loading a monitor. If this works, you need only save out the code in sections as we did with our program.

The program is THE FACTORY, Copyright 1983, Sunburst Communications.

TYPE OF PROTECTION: This program will read TRACK 18 SECTOR 2 for the ASCII values of SB. If present the program will run properly. This program appears to be totally unprotected in its original form, but the ASCII SB is the program's protection 'key'.

HOW TO COPY: Copy with any good copy program that will copy the entire disk. Alternatively, a file copy of the disk may be performed and the SB bytes placed on track 18 sector 2 at bytes 149 and 150 respectively.

HOW TO MAKE A WORKING COPY WITHOUT ERRORS ON THE DISK:

As you can see from above explanation the program does not require any errors on the copy disk. As like all other protected programs there is still a 'KEY' (in this case the SB) required on the disk that is necessary for the proper operation of the program. The 'KEY' may be anything on the disk. Bad blocks, modified sectors, extra tracks, track arcing, nybble counting, density changes, synchronized tracks and even just a couple of ASCII bytes on a sector are all forms of protection 'KEY's. The 'KEY' is the required element on a disk that the program checks for. Without the proper key the program will crash; with the proper key the program will execute normally. We are seeing a great deal of programs where the disk appears normal in most respects, yet there is still something on the disk that is required for the proper operation of the program. This type of protection may be very sophisticated or very easy. Just don't assume that the programmer is using some super new scheme when you encounter this type of protecion. Many times, if the program is just using the more basic types of schemes we find a B-P (BUFFER-POINTER) command used in combination with the protection scheme.

In this program, the checking is done in the program called "SBLSOGOB". This is a basic program that loads in a machine language program. LOAD "SBLSOGOB" and list LINE 600. This is where you will find the B-P. It is calling for a check of 18/2 and looking for an SB at byte 149 and 150. The program will act on the result of that check in LINE 15050. If the result is not equal to "SB" the program will be sent to a SYS that will crash the program.

Normally we would just change the <> to an =, but we must take care to keep the line length the same so that the machine language program that follows the BASIC resides in the proper location. An = will leave us one byte short in this line, causing the entire machine language section to be one byte too low.

1. Copy the disk with any good copy program that will copy the files only. Alternatively, copy the disk with a full disk copier and with a track and sector editor at TRACK 18 SECTOR 2, locate the SB. Change to HEX mode and change the 5342 (SB) to 0101.

2. LOAD "SBLSOGOB",8

3. Now LIST 15050. Our task is to alter the code while keeping the line length the same. Many protection schemes written in BASIC will check the length of the BASIC program to detect if the program was tampered with. It is important to maintain the same length of the program when modifying some BASIC protection schemes. We will accomplish this task by inserting a space in place of the < and changing the > to an =. The line should now read:

```
FORDL=1TO4000:NEXTDL:IFA$ ="SB"THENSYSAA
```

4. Save the altered code with SAVE"@0:SBLSOGOB",8

5. YOU'RE DONE!

Keep this technique in mind when unprotecting BASIC programs. You may even find programs that will execute a checksum to determine if the original code has been altered.

WIZARD ADDENDUM - WIZARD (C) 1984 by Progressive Peripherals & Software.

With the help of user-feedback we learned of another version of the Wizard program. We were advised by one of our users that the entry point for his version of the program was $10FC (SYS4348) rather than $198C. You may wish to try the procedure again with this entry point. The entire procedure can be found in the November NEWSLETTER. You may also wish to refer to the December Addendum for further clarification of the technique.

We have also learned that there seems to be a bit of confusion as to how and when to insert the cartridge-based machine language monitor. Once the protected program is running, insert your cartridge monitor with your CARTRIDGE SWITCH OFF. DO NOT insert a cartridge with the cartridge switch on. Remember, any time that you insert a cartridge into the computer with the power on you risk damage to the cartridge or computer. This is why we recommend the purchase of a cartridge board or expansion board.

The procedure for installing a cartridge switch is described in the PROGRAM PROTECTION MANUAL VOLUME I.

C.S.M. also sells a single slot cartridge board complete with a reset switch, GAME switch, EXROM switch, ENABLE switch and POWER switch.

PRODUCT REVIEW . . . . . .TRACKMIMIC(TM) BY A.I.D. CORP

TRACKMIMIC(TM) (hereafter referred to as TM) is a hardware and software product that is used to copy disks and/or specific tracks. TM contains a hardware device that replaces one of the VIA chips (6522) in the disk drive. This is accomplished by removing the chip from your disk drive and replacing it with a dip header and a ribbon cable connected to a circuit board. This circuit board plugs in the modem port of your computer. The circuit board supplied with TM appears to contain an onboard VIA chip and a '7400' series IC.

The people at A.I.D. CORP. would not release ANY details of how TM operates to us, so the information that we present here was obtained by our own inspection. The people at A.I.D CORP. could not (or would not) even tell us which chip that we had to replace in order to hook the unit up during our phone conversation with them (as you all know some of the chips in the drive are soldered in and cannot be easily removed). We were so disappointed with the information that we received during our phone conversation that decided not to buy a unit. We felt that if the folks at A.I.D. CORP. were not even willing to tell us a LITTLE about their product, there must be something wrong. Face it folks, when somebody is asking $89.95 (that's right $89.95) for a copy program, the company should be willing to tell you a little about what it does or what is necessary to make it work!!

Well, a good friend of mine saw the ad for TM and decided to order one ('AFTER ALL IT'S ONLY $89.95'). After working with the unit for about a week here is our evaluation of the TM disk back-up system.

First the good news......

1. TM is the fastest copy utility for the 1541 disk drive, appx. 2 minutes and six swaps of the disk. This product is quick.
2. TM will copy many different copy protection schemes without banging the head or rattling the drive.
3. TM is fully automatic and requires no user intervention to make a backup of many protected disks.
4. TM uses a unique approach to copying disks, this method is supposed to allow the user to examine the whole track in the computer's memory, sync marks and all.
5. TM appears to be able to copy all of the standard protection schemes that have been around for a few months or more, including extra tracks, duplicated sectors, and all the standard header errors (20, 21, 22, etc.).

Now the bad news........

1. Everything that TM successfully copied for us could be copied by other less expensive copy programs.
2. TM's function that allows the user to examine and modify a track was not reliable. We found that it would routinely report wrong information on the track that we were examining.
3. Even when TM would read data in properly from a track, it often would not reliably reproduce that data on the copy disk.
4. TM DID NOT copy any of the 'LATEST PROTECTION SCHEMES' that we examined including: TRACK ARCING, SYNCHRONIZED TRACKS, NYBBLE COUNTING, MODIFIED DENSITY, etc.
5. A few of the newer schemes that TM could NOT copy were easily copied by other less costly copy utilities.

We tried TM on many different protection schemes. Most of the protection schemes that have been around for a while were easily copied by TM, whereas the many of the newer, more sophisticated schemes were not copyable by TM.

In conclusion we feel that TM is not a very effective copy program when compared to what is already out on the market. We found a few programs that TM could not copy, yet these same programs were easily copied by other copy utilities. Hopefully the folks that produced TM will make some revisions and improvements to the program. The concept of hardware modifying the drive to make a more efficient copy program is interesting and probably is where the next generation of copy utilities will be coming from. For now I recommend that you save your money and wait for something better to come along.

HARDWARE LIMITATIONS OF THE 1541 DRIVE

We have seen many different program protection schemes used on the 1541 drive. We have even developed many of the protection schemes that are currently being used by a number of different software companies. We have researched the 1541 disk drive and found its physical limits. We know what can and what can not be duplicated on the 1541 disk drive. There are many different protection techniques that may be detected by the 1541, yet these same techniques are not able to be produced on the 1541. This fact is due to the physical limitations of the disk drive. More and more we are seeing protection schemes that capitalize on the hardware limits of the disk drive. These hardware limits will be the controlling factors as to whether or not the end user will be able to copy a disk or not.

Our business is program protection. We protect programs for other companies and we protect our own. We also teach people the how's and why's of program protection. We are not some pompous company that claims to have the 'TOP SECRET STUFF' on program protection (only to give you something less than you bargained for). We take the time to investigate many different protection schemes and most importantly, we share the knowledge that we have learned. It is through this time-consuming and laborious investigation process that we have gained our knowledge. Now on to the topic at hand.

Let's take a look at the various options that a programmer must consider when chosing a viable and effective program protection scheme.

First and foremost, there are no secrets to program protection. Each program is protected thru a logical and definable set of instructions. Each instruction will have a specific purpose and cause a specific task to be performed. The programmer is able to control which task is to be performed and in what order it is to be performed.

Second, the technique that the programmer choses must be something that is capable of being detected on the 1541 disk drive (i.e. nybble count, track arcing, synchronized tracks, extra tracks, modified formats etc.). It is important to note that the protection scheme chosen is one that must be detectable on the 1541 drive. It is not necessary or even desireable to have the scheme producable by the 1541 drive.

Third, the protection scheme chosen should not be easily reproduced on the 1541. If possible the scheme selected should not be able to be written on the 1541 under any circumstances. In other words, the scheme selected should take advantage of the hardware limitations of the 1541 drive.

The hardware limits of the drive include, but are not limited to, the following:

1. The small amount of RAM available in the 1541 drive (2K total). In order to accurately reproduce a track, large amounts of memory are sometimes required (e.g. data blocks longer than 1K).

2. The width of the R/W head on the 1541 is wide in comparision to what is required. This prevents the use of adjacent tracks and half-tracks.

3. The drive itself uses 4 separate densities during the normal course of operation. This allows the use of mixed densities on a single track, which is very difficult to copy.

4. Speed variations from disk to disk and from track to track on the same disk. This is a hinderance to nybble counting and nybble copying.

5. Speed variations also make track synchronization, track arcing and spiral tracking difficult to reproduce. As the speed varies it causes the relationship between tracks (and half-tracks) to vary.

6. The 1541 does not reference the timing hole on the disk. This compounds the problems of speed variation when moving the R/W head over large distances.

Let's now examine each of the hardware limits presented here in a little more detail. We will view each limitation as to how it applies the making a copy of the original disk.

1. LIMITED RAM... Approximately 1K of the 1541's RAM is required for internal housekeeping (stack, 0 page pointers, communications, etc.) and for the ML program used by a copy program to read from and write to the disk. This only leaves us with appx. 1K for a data storage buffer. When we read data in from the disk it is necessary to temporarily store it the drive's RAM memory before it is transferred over to the computer. If the original disk uses data blocks longer than 1K, most copy programs will have an extremely difficult time copying the disk.

When reading data from a disk the limited RAM is not a great hindrance. It is possible to sync up to a particular reference point, then to read 1K of memory and transfer it to the computer. Then all the copy program has to do is find this same reference point on the next revolution of the disk, skip over the first 1K and then read the second 1K of data from the disk. This process may be repeated until the whole track has been transferred to the computer's memory.

Writing data back out to the disk is where the real problem comes into play. It is necessary to write data out in 1K chunks (or less) due to the limited RAM in the drive. Data must be written to the disk in 'sync'. This means that the data must be timed out to the disk in a predictable and reliable pattern. The only way to maintain this predictable pattern is to write the data out in one continuous stream, with no interruptions. The way that the 1541 overcomes this problem of synchronizing the data is by keeping the total data block sector length to 330 GCR bytes (plus or minus). Each time that the data block is written to the disk, the sync mark and the data block identifier are also written out. This insures that the data will always be synchronized with its sync mark.

If we should write out 1K of data and then try to write out additional data in synchronization with the first chunk of data, there will be problems. The timing relationship of the drive and the disk prevent the data from maintaining the perfect relationship they had on the original disk. If the timing varies, even by one bit, the data of the second chunk will now be unintelligeable and the copy will fail. In practice the data will be found to have varied by more than just a few bits and by as many a dozen bytes from the proper orientation. Rotational speed of the disk also affects how well the data will match the original. Disks vary in rotational speed, from track to track and from disk to disk. The speed also varies during a single revolution of the disk. This variation is not predictable, nor is it reproduceable. The variance of speed plays an important part in how accurately the two chunks of memory match on the copy disk. In reality it is difficult, if not impossible, to maintain the relationship between 2 or more blocks of data written to the disk.

2. WIDTH OF THE R/W HEAD... Tracks on the 1541 drive are spaced at 48 tracks per inch (appx 0.020 in.). There are many commercial disk drives that use a track spacing of 96 tpi (0.010 in. appx). Since the tracks are further apart on the 1541 each track may be wider than tracks written on the 96 tpi drive. Due the width of the R/W head of the 1541 drive it is not possible to write adjacent tracks and half tracks side by side.

If an original disk is prepared on a drive that uses the 96 tpi drive, it is possible to write adjacent tracks and half tracks. Each track will be narrower than the standard 1541 track, but they will be readable if done properly.

3. DENSITIES... The 1541 drive uses 4 different densities to write data on every disk. The density refers to the time between bits on the disk. The outer tracks will time bits out at a higher rate than the inner tracks. The higher density allows the 1541 drive to store more data on the outer tracks. Normally, density is set to a specific value for a given track and this density is not changed on the track. If the programmer should change density in the middle of the track it would be next to impossible to detect and reproduce the density changes. If the programmer used multiple density changes and combined this with various other techniques the difficulty level would be tremendously increased.

4. SPEED VARIATIONS OF THE DRIVE... As seen above, speed variations of the drive can be a hinderance to copy programs. Speed will have a direct effect upon the total number of bytes written to the track, the density of the bytes written and the relationship between sectors on a track. If the speed varies by just 1% on a track this could throw off the nybble count by as much as 77 bytes. This means that the total track length could be 77 bytes longer or 77 bytes shorter than the original track. During the normal course of operation the drive will compensate for the variation in speed from disk to disk by varying the length of the gap bytes. In a nybble count of a track it is not important which bytes are gap bytes or which are data bytes; they are all counted. This assumes that no other changes have been made to the track such as density or extra long data blocks. (note: it becomes even more difficult to copy the disk when multiple schemes are employed).

5. SYNCHRONIZED TRACKS... Each track has a beginning and an ending. If we use sector 0 as the beginning and sector 20 as the end of a track we can begin to establish our track to track relationship. The 1541 drive writes data out to the disk during the format process in a somewhat random procedure. This means that each and every track could start any where on the disk in relation to the timing hole (we are using the timing hole only as a handy reference point on the disk - the 1541 does not use the timing hole for any purpose whatsoever). If the original disk was created on a programmable drive, each and every track could be synchronized to the timing hole, thereby insuring the proper synchronization from track to track.

While it is possible to time each track and establish the relationship from track to track and then synchronize one track to another, it gets very difficult to synchronize multiple tracks. Track arcing and spiral tracking are variations of synchronized tracks whereby relatively small amounts of data are written to the disk on adjacent tracks and half-tracks. The reason that track arcing and spiral tracking works is due to the fact that at no time is the data on a track and its adjacent half-track side by side. For instance, if the programmer wrote 1/4 of a track and then stepped the head a half track and continued writing data for anothe 1/4 of the track the information would be staggered on the disk. This staggering of data is what allows these schemes to function properly.

The next time that you try to copy one of your 'latest and greatest' programs and you find that there is not any copy utilities that will copy them, you won't have to wonder why. Copy protection has reached the physical limits of 1541 drive.

What about hardware-modifying the drive to overcome some of these limitations of the 1541?? Well, this has been tried, mostly without much success (see the product review of the TRACKMIMIC(TM) elsewhere in this issue). The limiting factor here is that 1541 has so many hardware limitations that it is not practical to try to overcome all of them.

IMPORTANT NOTICE.....

Folks, if you buy one of our competitor's products and are not happy with it, DON'T complain to us. Unfortunately, we can not help you when you buy what seems to be a similiar product from another company. During the past few months we have received many complaints from people about other cartridge backup systems. There has been a tremendous surge in companies trying to jump on the cartridge backup band wagon (four different companies that we know of). Don't be misled by confusing claims and advertising. Check out your purchases carefully. If you are not sure of something, ask specific questions.

SPECIAL NOTICE....... READ THE ADVERTISING BEFORE YOU PURCHASE YOUR NEW 128K COMPUTER!!!! THERE ARE A FEW COMPANIES THAT ARE OFFERING THE B-128 COMPUTERS FOR SALE.. THE C-128 COMPUTER IS MODEL THAT YOU WILL WANT TO PURCHASE. THE C-128 IS THE TRIPLE PROCESSOR MACHINE THAT LOOKS TO BE A WINNER. AS FAR A WE KNOW ABSOLUTELY NO DEALERS HAVE THE C-128 MACHINE IN STOCK!!!! DON'T BE FOOLED BY LOOK ALIKE ADVERTISING GIMMICS!!!

SYMBOLIC DISASSEMBLERS

An interesting new type of utility that we have seen lately is a 'SYMBOLIC DISASSEMBLER'. This type of utility takes an OBJECT (machine language) program and converts it to a SOURCE (symbolic) code program. The source code produced is not identical to the original source code, of course, but it is close enough to allow you to edit and modify the program much more easily than with just a machine language monitor. This is possible because the source code produced is compatible with common assembler packages. The power that this gives you is considerable.

For example, when editing a program with just an ML monitor, it is easy enough to change a few bytes here and there. If you want to insert a section of code, however, it is much harder. You have to move the old code down to make room, edit it so it will work in its new location, etc. If you find out you need a little more room when you're done, you'll have to go through it all again.

With an assembler, editing a source code program is more like editing a BASIC program. You can insert and delete statements easily. You can give symbolic names to important memory locations, much like you use variables in BASIC. Many assemblers give you programming aids such as find, replace and renumber commands to make editing even easier. You can usually even relocate the entire program somewhere else in memory with a minimum of trouble. Another advantage often overlooked is that you can add comments to the source code as you go through it. All of these advantages are available to you if you can disassemble your code into source file form.

The first good symbolic disassembler we've seen is the SYMBOL MASTER (tm) from Schnedler Systems, 1501 N. Ivanhoe, Dept. R3, Arlington, VA 22205; phone (703) 237-4796. To use SYMBOL MASTER (tm), you must first examine the program being disassembled and decide what is actual code versus what is simply data tables. Armed with this information, the system can produce source code files compatible with MAE, PAL, Commodore and Develop-64 assemblers.

SYMBOL MASTER (tm) is a good program with several advanced features. For example, it can be set to automatically use the names of KERNAL routines and operating system variables when it encounters them. It generates labels for each memory location used and recognizes locations that are outside the program being disassembled. It can even ferret out code hidden in BIT instructions, which is a common way to reduce program space as well as hide program lines.

Although we haven't tested it exhuastively, it appears to do everything it claims. We did not encounter any bugs or problems with it, and the documentation was thorough. The only potential drawback we noticed was that the object program being disassembled must be able to fit into memory with the disassembler system.

If you have an immediate need for a symbolic disassembler, SYMBOL MASTER (tm) should be more than satisfactory. However, we have recently received news of another such system, which we will be reviewing next month. Among other things, it can disassemble an object program from disk or memory. If you can afford to wait, you may want to see what we have to say about it before making up your mind.

ULTRA LIGHTNING

CSM Software Inc. is proud to present the ULTRA LIGHTNING LOADER. This fantastic utility speeds up your disk drive at least 500%. Unlike similar utilities on the market, ULTRA LIGHTNING is totally compatible with all current Commodore 64 software INCLUDING COPY PROGRAMS AND QUICK-LOAD UTILITIES! This means your nibble or fast copy programs will now be 5 times faster. If you already have a quick-load type utility ULTRA LIGHTNING will speed it up even more! This can give you an increase in speed of up to 25 times the normal 1541 speed.

ULTRA LIGHTNING is a special treat for our newletter subscribers, at no charge. I'm sure you'll agree it is worth many times its cost. To put ULTRA LIGHTNING to use on your computer, follow this procedure:

1. Type in the program listed below. Be especially careful to type the data statements correctly.
2. When you are finished typing the program, DO NOT try to run it. Save the program to a disk first.
3. Turn off the computer and disk drive.
4. Turn the drive back on first, then the computer.
5. Load in the ULTRA LIGHTNING program and run it. The program will check the data statements and tell you if you have made any errors.
6. If there are no errors, the program will ask you for your drive's device number.
7. The program will then download a special routine into the disk drive, and also patch itself into the computer's operating system.
8. When it is finished, you are ready to load your copy utility or other program. Don't attempt to examine the code in memory or execute any statements other than a LOAD statement.
9. That's all there is to it.

```
1 CK=0 :FOR I=1 TO 134 :READ A :CK=CK+A :NEXT :RESTORE
2 IF CK<>19048 THEN PRINT "ERROR IN DATA STATEMENTS" :END
10 PRINT CHR$(147) TAB(9) "ULTRA LIGHTNING LOADER" :PRINT :PRINT
20 INPUT "ENTER DISK DEVICE NUMBER"; DN :IF DN<8 OR DN>11 THEN 20
30 PRINT :PRINT :PRINT "INITIALIZING - PLEASE WAIT ..." :PRINT
40 OPEN 15,DN,15,"IO"
50 OPEN 2,DN,2,"#"
60 PRINT#15, "M-W" CHR$(29) CHR$(0) CHR$(1) CHR$(DN)
70 PRINT#15, "M-W" CHR$(35) CHR$(0) CHR$(1) CHR$(73)
80 PRINT#15, "M-W" CHR$(0) CHR$(4) CHR$(1) CHR$(99)
90 FOR I=1 TO 32 :READ A :PRINT#2,CHR$(A); :NEXT
100 CLOSE 2 :CLOSE 15
110 FOR I=0 TO 101 :READ A :POKE 52992+I,A :NEXT
120 SYS 52992
130 PRINT "ALL DONE. NOW LOAD YOUR PROGRAM" :NEW
140 DATA 16,129,43,23,193,34,240,54,117,30,44,152,99,32,202,15
150 DATA 63,219,55,19,233,59,172,70,199,41,71,210,13,55,151,47
160 DATA 169,15,162,255,143,48,3,169,207,162,255,143,49,3,96,169
170 DATA 4,141,32,208,141,33,208,160,63,191,38,207,73,170,32,202
180 DATA 241,136,208,245,240,254,139,139,235,226,233,254,229,237
190 DATA 53,138,138,138,138,138,138,138,138,138,138,138,138,138
200 DATA 138,138,138,187,187,167,249,141,230,229,229,236,138,230
210 DATA 227,248,250,235,52,138,138,138,138,138,138,138,138,138
220 DATA 138,138,138,138,138,187,187,187,187,187,187,187,187,57
```

ALL ITEMS IN STOCK FOR IMMEDIATE SHIPMENT

PROGRAM PROTECTION MANUAL FOR THE C-64 (VOLUME II) $34.95

THIS MANUAL BEGINS WHERE THE FIRST ENDED. COVERS THE LATEST ADVANCES IN PROGRAM PROTECTION, INCLUDING ENCRYPTION, GCR CODING, UNDOCUMENTED OPCODES, ADVANCED CARTRIDGE PROTECTION SCHEMES, AND MUCH MORE.

1541 DISK DRIVE ALIGNMENT PROGRAM 44.95

See Oct. 84 Compute! Gazette for review

PROGRAM PROTECTION MANUAL FOR THE C-64 (VOLUME I) 29.95

CARDO 5 SLOT EXPANSION BOARD FOR THE C-64 65.00
INCLUDES A RESET BUTTON AND THE CARTRIDGE SWITCH

ADVANCED ML BOOK FOR THE C-64	(ABACUS)	19.95
ANATOMY OF THE 64	(ABACUS)	19.95
ANATOMY OF THE 1541	"	19.95
MACHINE LANGUAGE BOOK OF THE 64	"	14.95
TRICKS AND TIPS FOR THE 64	"	19.95

1541 MAINTENANCE MANUAL (GOSUB) 29.95

PROMEMADE 99.50

THIS IS THE EPROM PROGRAMMER THAT WE USE HERE!! USE IT TO MAKE YOUR OWN OPERATING SYSTEM CHIPS, CARTRIDGE PROGRAMS OR A NEW DOS FOR THE DISK DRIVE. SOFTWARE INCLUDED ON DISKETTE.

DATARASE 34.95

This will erase all U.V. erasable EPROMS. A compact and easy to use unit that will erase 2 EPROMs in 15 minutes or less.

BOARDS, SOCKETS AND EPROMS:

PCC2	2 CHIP CARTRIDGE BOARD	6.95
PCC4	4 CHIP BANK SWITCHED CARTRIDGE BOARD	19.95
PCVH2	PLASTIC HOUSING FOR ABOVE BOARDS	2.95
LP28	28 PIN SOCKET FOR EPROM	.59
ZIF28	ZERO INSERTION FORCE 28 PIN SOCKET	4.75
2732	EPROM (USE IN CARTRIDGE 4K)	5.00
2764	EPROM (USE IN CARTRIDGE 8K OR IN DISK DRIVE W/AD)	7.50
27128	EPROM (16 K ON ONE CHIP - USE IN PCC4)	15.95
68764	8K PIN FOR PIN REPLACEMENT FOR ROM's	29.95
AD	USE THIS ADAPTER TO USE THE 2764 EPROM IN THE DISK DRIVE OR IN THE COMPUTER. IF YOU DON'T USE THE ADAPTER USE 68764 EPROM	8.95

HESMON (R) ML MONITOR ON A CARTRIDGE. 25.00

This version also includes a DOS WEDGE and PROMOS (PROMENADE) software on the cartridge.

* *

BACK IN STOCK!!!! INSIDE COMMODORE DOS $19.95
BY RICHARD IMMERS & GERALD C. NUEFELD
We know that there are many people that have been waiting for this book to become available.... Well, here it is!!

* *

CARTRIDGE BACKER $54.95

THIS WILL 'BACK UP' 99% OF ALL CARTRIDGES ON THE MARKET TODAY! IT WILL DOWN LOAD THE NEW VERSION OF THE PROGRAM TO DISK AND WRITE AN AUTO BOOT LOADER FOR IT. AS MANY AS TEN CARTRIDGES MAY BE STORED ON A DISK. IT WILL ALSO TELL YOU WHAT CODE IT CHANGED IN ORDER TO 'BACK THEM UP'. THE UNIT CONTAINS IMPROVED SOFTWARE THAT WILL ALLOW YOU BACK UP ALMOST EVERY CARTRIDGE AVAILABLE. IT WILL ALSO BACK UP MANY DISK PROGRAMS (ARTS, ACTIVE, PARKERS AND MORE). THE PROGRAM ALSO CONTAINS SUPER SAVER - DISK TO DISK AND DISK TO TAPE FILE COPYING.

By the way, folks, CARTRIDGE BACKER does not require the use of an 'EXTERNAL RAM BOARD' in order to run the disk version of the program, as do some of the other cartridge backup utilities on the market.

* * *

SINGLE SLOT EXPANSION BOARD WITH RESET BUTTON 24.50
A HIGH QUALITY BOARD WITH GAME, EXROM, ENABLE AND POWER SWITCHES AND A RESET BUTTON BUILT IN. PERFECT FOR THE USER WHO DOESN'T NEED A $65.00+ EXPANSION BOARD.

XETEC PRINTER INTERFACE 54.95
THIS INTERFACE WILL TURN YOUR GEMINI, DELTA OR EPSON PRINTER INTO A CORRESPONDENCE QUALITY PRINTER!!!

DISK DRIVE BELT -- 10.00
This is the belt that spins the disk.

* * *

PROGRAM PROTECTION MANUAL -- VOLUME II (AVAILABLE NOW) 34.95

THIS MANUAL STARTS WHERE THE FIRST LEFT OFF... A FULL INTRODUCTION TO MACHINE LANGUAGE, RESETS, INTERRUPTS AND AUTO BOOTS. IT ALSO COVERS THE MOST RECENT ADVANCES IN PROGRAM PROTECTION. HALF-TRACKS, EXTRA TRACKS AND SECTORS, NON-STANDARD FORMATS, NIBBLE COUNTING, TRACK ARCING, UNIMPLEMENTED OPCODES, ENCRYPTION TECHNIQUES AND MUCH, MUCH MORE!!! EXPERTS HAVE AGREED TO CONTRIBUTE SELECTED CHAPTERS IN THEIR OWN FIELD OF EXPERTISE. THE INFORMATION IS PRESENTED IN AN EASY-TO-UNDERSTAND FASHION FOR THE BEGINNERS AND EXPERTS ALIKE. SAMPLE PROGRAMS ARE INCLUDED TO HELP YOU UNDERSTAND ALL THE PRINCIPLES.

Sorry for the delay in getting the PPM volume II to press folks. At the last minute we decided to add an additional 50 pages to the manual - over 275 pages in total. Well, the book is hot off the presses and in stock now!!!

P.S. THIS IS A NEW MANUAL, NOT JUST A 3rd EDITION.

C.S.M. SOFTWARE INC.
P. O. BOX 563
CROWN POINT, IN. 46307
219-663-4335

all items plus shipping - $3.50 for most items